

## Prototypy praktických řešení pomocí Ajaxu

Masivní rozmach Ajaxu a zejména stále širší povědomí a zájem o tyto technologie nastal nikoliv po Garretově článku, ve kterém byl pojem Ajax poprvé použit a byly popsány jeho základní principy, ale až potom, co se na webu objevily zajímavé a užitečné aplikace, které tyto principy využívaly. Uživatelé si tak mohli v praxi vyzkoušet, jaké interaktivity a uživatelského komfortu je možné u webových aplikací dosáhnout, a logicky začali tuto funkcionalitu požadovat i u jiných webových aplikací.

### Využití internetových služeb Google

Začněme možnostmi využití funkcionality populárních služeb společnosti Google, především vyhledávače a Google Maps.

#### Ajaxová aplikace využívající vyhledávač Google

Funkcionalitu populárního vyhledávače Google je možné zabudovat i do vlastní aplikace Ajax. Pro implementování vyhledávání využijeme aplikační rozhraní Google Ajax Search API (<http://code.google.com/apis/ajaxsearch/>).

Když chceme vytvořit aplikaci Ajax na základě tohoto rozhraní, nejdříve musíme zavést modul Google Ajax Search API.

```
<script src="http://www.google.com/uds/api?file=uds.js&v=1.0"
  type="text/javascript">
</script>
```

Moduly se zavádějí pomocí funkce API:

**google.load(moduleName, moduleVersion, optionalSettings)**

V naší aplikaci chceme využívat interaktivní vyhledávání. Proto zavedeme modul:

```
<script type="text/javascript">
  google.load("search", "1");
</script>
```

Pro vyhledávání využijeme funkce API:

```
var searchControl = new google.search.SearchControl();
searchControl.addSearcher(new google.search.WebSearch());
searchControl.draw(document.getElementById("searchcontrol"));
```

Kompletní kód vyhledávací aplikace Ajax bude:

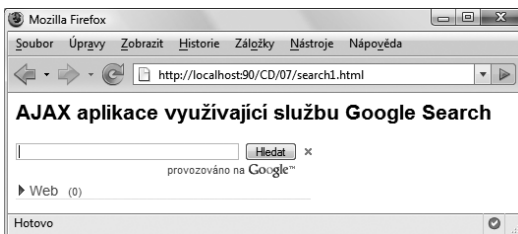


07/  
search1  
.html

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html;
  charset=windows-1250">
  <script src="http://www.google.com/jsapi?key=ABCDEFGF"
  type="text/javascript"></script>

  <script type="text/javascript">
    google.load("search", "1");

    function initialize()
    {
      var searchControl = new google.search.SearchControl();
      searchControl.addSearcher(new google.search.WebSearch());
      searchControl.draw(document.getElementById("searchcontrol"));
    }
    google.setOnLoadCallback(initialize);
  </script></head>
<body>
  <h2> AJAX aplikace využívající službu Google Search </h2>
  <div id="searchcontrol"></div>
</body>
</html>
```



**Obrázek 7.1:** Nejjednodušší stránka Ajax využívající službu Google Search

Zatím vyhledáváme jen na webu. Do aplikace můžeme přidat další funkce pro lokální vyhledávání a vyhledávání videozáznamů, obrázků a vyhledávání na blog-stránkách, v novinkách a v knihách.

```
..
searchControl.addSearcher(new GlocalSearch());
searchControl.addSearcher(new GvideoSearch());
searchControl.addSearcher(new GblogSearch());
searchControl.addSearcher(new GnewsSearch());
searchControl.addSearcher(new GimageSearch());
searchControl.addSearcher(new GbookSearch());
..
```

Kompletní kód aplikace pro vyhledávání pomocí více typů vyhledávačů bude:

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html;
    charset=windows-1250">
  <title>Příklad pro Google AJAX Search API </title>
  <link href="http://www.google.com/uds/css/gsearch.css"
    type="text/css" rel="stylesheet"/>

  <script src="http://www.google.com/uds/api?file=uds.js&v=1.0"
    type="text/javascript">
  </script>

  <script language="Javascript" type="text/javascript">
  //
    function OnLoad()
    {
      var searchControl = new GSearchControl();
      var localSearch = new GlocalSearch();
      searchControl.addSearcher(localSearch);
      searchControl.addSearcher(new GwebSearch());
      searchControl.addSearcher(new GvideoSearch());
      searchControl.addSearcher(new GblogSearch());
      searchControl.addSearcher(new GnewsSearch());
      searchControl.addSearcher(new GimageSearch());
      searchControl.addSearcher(new GbookSearch());

      // Bod pro localSerch
      localSearch.setCenterPoint("Praha");

      searchControl.draw(document.getElementById("searchcontrol"));

      // Heslo pro vyhledávání
      searchControl.execute("ajax");    }
      GSearch.setOnLoadCallback(OnLoad);
    //]]&gt;
&lt;/script&gt; &lt;/head&gt;
&lt;body&gt;
  &lt;h2&gt;AJAX aplikace využívající službu Google Search&lt;/h2&gt;
  &lt;div id="searchcontrol"&gt;Načítá se&lt;/div&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>
</div>
<div data-bbox="896 121 957 146" data-label="Image">
<img alt="CD-ROM icon"/>
</div>
<div data-bbox="897 146 956 187" data-label="Text">
<p>07/<br/>search<br/>.html</p>
</div>
<div data-bbox="922 759 985 860" data-label="Page-Footer">
<p>Prototypy prak-<br/>tických řešení<br/>pomocí Ajaxu<br/>7</p>
</div>
```



**Obrázek 7.2:** Stránka Ajax využívající službu Google Search

## Ajaxová aplikace využívající Google Maps

Google maps je bezplatná, online mapová služba společnosti Google. Nabízí posouvateľné mapy a satelitní snímky celého světa a plánovač cest s vyhledáváním a lokalizováním firem. Označení „satelitní snímky“ není přesné, protože část záběrů s největším rozlišením je z leteckého snímkování. „Pravé“ satelitní snímky pocházejí převážně z komerčního satelitu QuickBird s maximálním rozlišením 60 cm na pixel. Služba Google Maps obsahuje výsledky vyhledávání (například výpisy podniků, obrázky a související informace), které jsou poskytovány od nezávislých dodavatelů.

Službu je možné využívat jen v souladu se smluvními podmínkami. Pro jednotlivce je služba Google Maps (včetně místních výsledků vyhledávání, map a fotografického obrazového obsahu) poskytována jen pro osobní a nekomerční použití. Pokud je uživatelem podnik, služba Google Maps může být používána jen uvnitř podniku a nemůže být komerčně poskytována. Službu můžeme volat z vlastní stránky HTML. Musíme si však vygenerovat klíč. Klíč pro používání API je možné po souhlasu s licenčními podmínkami vygenerovat na adrese:

<http://code.google.com/apis/ajaxsearch/signup.html>

V příkladu jsme použili zeměpisné souřadnice Prahy:



07/  
mapa  
.html

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html;
    charset=windows-1250">
</head>
<body>
  <h2>HTML aplikace využívající službu Google Maps</h2>
  
</body>
</html>
```

Nejjednodušším způsobem, jak využít službu Google Maps ve vlastní stránce Ajax, je využití aplikačního rozhraní Maps. Toto rozhraní se přilinkuje do aplikace:

```
<script type="text/javascript"
  src="http://www.google.com/jsapi?key=ABCDEFGH">
</script>
```

Moduly se zavádějí pomocí funkce API:

### **google.load(moduleName, moduleVersion, optionalSettings)**

V naší aplikaci chceme využívat interaktivně zobrazení mapy a vyhledávání. Proto zavedeme moduly:

```
<script type="text/javascript">
  google.load("maps", "2");
  google.load("search", "1");
</script>
```

Nejdříve nastavíme bod v zeměpisných souřadnicích, který bude ve středu zobrazené mapy, její měřítko a typ.

### **setCenter(center, zoom?, type?)**

Zoom se nastavuje v rozmezí hodnot 0 (celá zeměkoule) až 19, což je nejvyšší rozlišení.

```
map.setCenter(new google.maps.LatLng(50.09193, 14.41935), 10);
```

Kompletní kód aplikace bude:

```
<html>
<head>
<meta http-equiv="content-type" content="text/html;
  charset=windows-1250"></head>

<script type="text/javascript"
  src="http://www.google.com/jsapi?key=ABCDEFGH"></script>

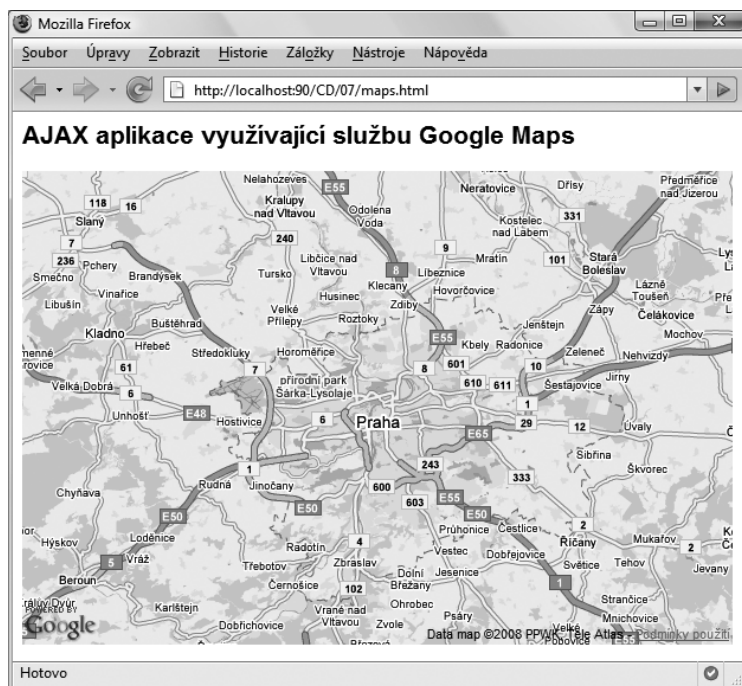
<script type="text/javascript">
  google.load("maps", "2");

  // Funkce se volá při načítání stránky
  function initialize()
  {
    var map = new google.maps.Map2(document.getElementById("map"));
    map.setCenter(new google.maps.LatLng(50.09193, 14.41935), 10);
  }

  google.setOnLoadCallback(initialize);
</script>
</head>
<body>
  <h2>AJAX aplikace využívající službu Google Maps</h2>
  <div id="map" style="width: 600px; height: 400px"></div>
  <div id="searchcontrol"></div>
</body>
</html>
```



07/  
maps  
.html



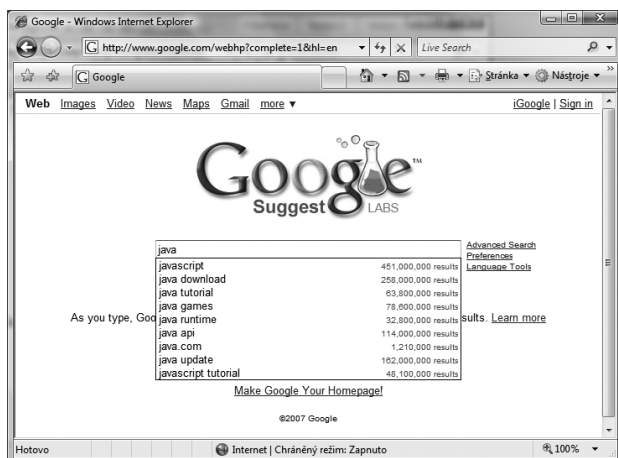
**Obrázek 7.3:** Nejjednodušší stránka Ajax využívající službu Google Maps

## Google Suggest

Populární vyhledávač Google se neustále vyvíjí a v době psaní této publikace byla v testovacím provozu funkce „Suggest“. Funkce během zadávání vyhledávací fráze doplňuje nejčastěji zadávané výrazy, přičemž ke každému nabízenému výrazu se zobrazuje i počet výsledků vyhledávání. Propracovaný algoritmus tedy během psaní nabídne ta slova, která se mezi populárními nejvíce podobají vašemu zadávanému slovu. Není těžké uhádnout záměr tvůrců – chtějí nabídnout uživateli komfort a inspirovat ho textovými řetězci tak, aby strávil při používání služeb vyhledávače více času, takže se uživateli zobrazí více sponzorovaných odkazů, což naplňuje rozpočet provozovatele.

Provázání tvorby vyhledávacího dotazu se statistikami a strategickými záměry Google vypadá na první pohled velmi zajímavě. Když se záměry provozovatele soustředí výhradně na podsouvání textových řetězců, pomocí nichž dosáhli uživatelé nejlepší výsledky, vše je v absolutním pořádku. Když však začnou podsouvané texty navigovat uživatele k reklamním nabídkám, bude to špatné.

Samotná funkce Google Suggest pracuje na stejném principu jako funkce „Did you mean“ (měli jste na mysli), když se vám podaří udělat překlep ve vaší vyhledávací frázi anebo když nějaká jiná fráze obsahuje více výsledků. Na rozdíl od „Did you mean“, které funguje klasicky, znovu načte obsah stránky, služba „Suggest“ využívá funkcionalitu Ajax a doplňuje vyhledávané fráze interaktivně bez nového načtení stránky. Testovací provoz běží na adrese <http://labs.google.com/suggest>.



Obrázek 7.4: Služba Google Suggest

## Volání služby Google Suggest

Doplňování funguje tak, že při každém stisknutí je na pozadí stránky HTML volán požadavek na server s adresou `http://www.google.com/complete/search?hl=en&js=true&qu=cast_vyhledavaci_fraze`. Jako parametr `qu` slouží už napsaná část vaší vyhledávací fráze. Vrácený výsledek je javascriptové pole, jehož obsah se zobrazuje pod vyhledávacím.

Když například zadáme jako část vyhledávací fráze slovo „java“, bude adresa URL ve tvaru:

`http://www.google.com/complete/search?hl=en&js=true&qu=ajax`

Na stránce se zobrazí výsledek v podobě obsahu javascriptových polí. Konkrétně pro zadaný výraz „ajax“ budou vrácena pole s tímto obsahem:

```
sendRPCDone(frameElement, "ajax",
  new Array("ajax", "ajax tutorial", "ajax examples", "ajax tutorials",
    "ajax
      toolkit", "ajax framework", "ajax asp.net", "ajax control
      toolkit",
    "ajax php", "ajax example"),
  new Array("67,800,000 results", "30,000,000 results", "4,680,000 results",
    "45,800,000 results", "1,610,000 results", "44,400,000 results",
    "9,780,000 results", "1,210,000 results", "89,300,000 results",
    "19,600,000 results"),
  new Array("")
);
```

Když si pozorněji prohlédnete vrácený text, jedná se vlastně o javascriptový kód, konkrétně o volání funkce:

**sendRPCDone(nepouzite, vyhledavany\_vyraz, pole\_vyrazu, pole\_poctu\_vyskyty, nepouzite\_pole)**

Proto napíšeme tuto funkci tak, abychom využili její automatické volání pro výpis serverem poskytnutých údajů. Nejjednodušší bude využít pro jejich výpis tabulku.

```
function sendRPCDone(nic, vyraz, polePojmu, poleVysledku, nicPole)
{
    var tabulka = "<table>";

    if (poleVysledku.length != 0)
    {
        for (var nC = 0; nC < poleVysledku.length; nC++)
        {
            tabulka += "<tr><td>" + polePojmu[nC] +
                '</td><td>' + poleVysledku[nC] + "</td></tr>";
        }
    }
    text += "</table>";
    var napovedaDiv = document.getElementById("napovedaDiv");
    napovedaDiv.innerHTML = text;
}
```

Funkce vypíše výsledky pro zadaný výraz (například ajax) ve tvaru:

ajax	67,800,000 results
ajax tutorial	30,000,000 results
ajax examples	4,680,000 results
ajax tutorials	45,800,000 results
ajax toolkit	1,610,000 results
ajax framework	44,400,000 results
ajax asp.net	9,780,000 results
ajax control toolkit	1,210,000 results
ajax php	89,300,000 results
ajax example	19,600,000 results

Funkci můžeme vylepšit tak, že do prvního sloupce tabulky umístíme místo prostého textu hledaného výrazu odkaz na klasický vyhledávač Google.

Doplňování funguje tak, že při každém stisknutí je na pozadí stránky HTML volán požadavek na server s adresou <http://www.google.com/search?q=> vyhledávaný výraz.

```
function sendRPCDone(nic, vyraz, polePojmu, poleVysledku, nicPole)
{
    var text = "<table>";

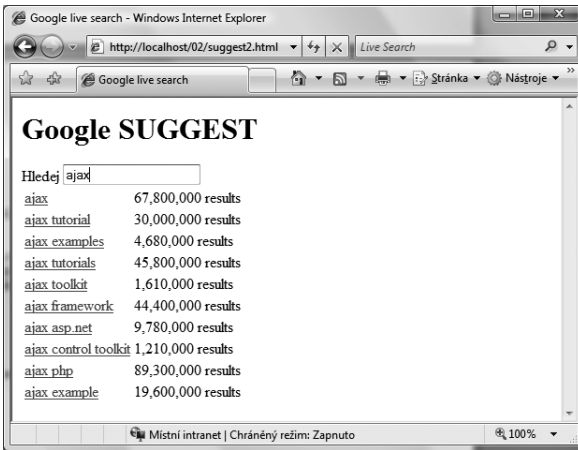
    if (poleVysledku.length != 0)
    {
        for (var nC = 0; nC < poleVysledku.length; nC++)
        {
            text += "<tr><td>" +
                "<a href='http://www.google.com/search?q=" +
                polePojmu[nC] + "'>" + polePojmu[nC] +
                '</a></td><td>' + poleVysledku[nC] + "</td></tr>";
        }
    }

    text += "</table>";

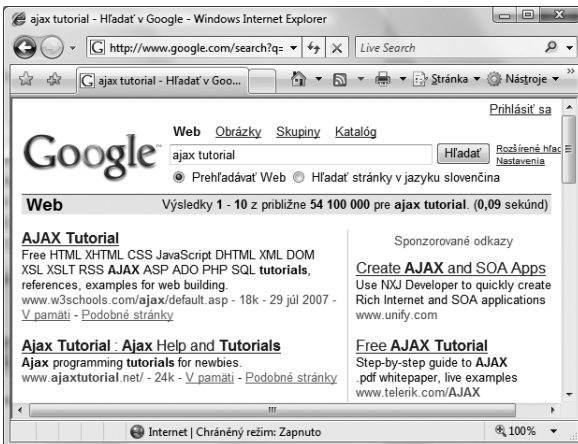
    var napovedaDiv = document.getElementById("napovedaDiv");
```



```
napovedaDiv.innerHTML = text;
}
```



Obrázek 7.5: Výpis údajů ve formě tabulky spolu s odkazem na každý nalezený termín



Obrázek 7.6: Klepnutím na doporučený výraz se zobrazí výsledky vyhledávání

Trochu jsme předběhli a na obrázcích ukázali výsledek, tedy způsob, jakým se budou zobrazovat výrazy interaktivní nápovědy. Vraťme se k budování stránky HTML využívající funkcionality Google Suggest. V těle stránky zobrazíme jen nadpis a textové pole pro zadávání vyhledávaného výrazu.

```
<body>
  <h1>Google SUGGEST</h1>
  Hledej <input id = "vyraz" type = "text"
  name = "vyraz" onkeyup = "Napoveda(event)">
  <div id = "napovedaDiv"><div></div></div>
</body>
```

Jako reakce na událost `onkeyup` v poli pro zadávání textového řetězce se volá funkce `Napoveda`. V jejím těle musíme detekovat text dosud napsaný do textového pole. Kvůli kompatibilitě prohlížečů musíme v některých případech místo události odeslané jako parametr využít událost `window.event`. Aby byl kód co nejjednodušší, využijeme podmínku typu:

```
proměnná = (podmínka) ? prirazeni_true: prirazeni_false;
```

Konkrétně v našem případě:

```
udalost = (udalost) ? udalost: window.event;
```

Dosud napsaný text přiřadíme do proměnné také pomocí podmínky zohledňující nekompatibilitu prohlížečů:

```
txt = (udalost.target) ? udalost.target : udalost.srcElement;

function Napoveda(udalost)
{
    udalost = (udalost) ? udalost: window.event;
    txt = (udalost.target) ? udalost.target : udalost.srcElement;

    if (udalost.type == "keyup")
    {
        if (txt.value) {getData("suggest.php?qu=" + txt.value);}
        else
        {
            var napovedaDiv = document.getElementById("napovedaDiv");
            napovedaDiv.innerHTML = "<div></div>";
        }
    }
}
```

Z bezpečnostního hlediska není povoleno posílat dotaz na stránky z domén třetích stran. Když adresa URL absolutní cesty k přenášenému dokumentu směřuje do jiné domény volání metody `open`, vyvolá výjimku. Proto vytvoříme ve vlastní doméně skript PHP, který zprostředkuje volání adresy URL v jiné doméně.

`suggest.php`

```
<?php
    $stream = fopen("http://www.google.com/complete/search?hl=en&js=true&qu="
        . $_GET["qu"], "r");
    while (!feof($stream))
    {
        $text = fgets($stream);
        echo $text;
    }
    fclose($stream);
?>
```

Kód můžeme vyzkoušet například pomocí parametru `ajax` v adrese URL `http://localhost/04/suggest.php?qu=ajax`.

Kompletní kód stránky HTML pro volání služby Google Suggest na pozadí a pro interaktivní zobrazování výsledků bez znovunačtení stránky bude vypadat následovně:

suggest.html

```
<html>
<head>
<meta http-equiv="content-type" content="text/html;
  charset=windows-1250">
<title>Google SUGGEST</title>
<script language = "javascript">

var xHttp= false;

function ZiskejData(zdrojDat)
{
  xHttp = VytvorXMLHttp();
  xHttp.open("GET", zdrojDat);
  xHttp.onreadystatechange = function()
  {
    if (xHttp.readyState == 4 && xHttp.status == 200)
      { eval(xHttp.responseText);}
  }
  xHttp.send(null);
}

function Napoveda(udalost)
{
  udalost = (udalost) ? udalost: window.event;
  txt = (udalost.target) ? udalost.target : udalost.srcElement;

  if (udalost.type == "keyup")
  {
    if (txt.value) {ZiskejData ("suggest.php?qu=" + txt.value);}
    else
    {
      var napovedaDiv = document.getElementById("napovedaDiv");
      napovedaDiv.innerHTML = "<div></div>";
    }
  }
}

function sendRPCDone(nic, vyraz, polePojmu, poleVysledku, nicPole)
{
  var tabulka = "<table>";

  if (poleVysledku.length != 0)
  {
    for (var nC = 0; nC < poleVysledku.length; nC++)
    {
      tabulka += "<tr><td>" +
        "<a href='http://www.google.com/search?q=" +
        polePojmu[nC] + "'>" + polePojmu[nC] +
        "</a></td><td>" + poleVysledku[nC] + "</td></tr>";
    }
  }
}
```



07/  
Suggest.  
html

```
    }
    tabulka += "</table>";
    var napovedaDiv = document.getElementById("napovedaDiv");
    napovedaDiv.innerHTML = tabulka;
}

function VytvorXMLHttp()
{
    var xmlHttp= false;

    if (window.ActiveXObject)
        {xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");}
    else if (window.XMLHttpRequest)
        {xmlHttp = new XMLHttpRequest();}
    return xmlHttp
}

</script>
</head>

<body>
<h1>Google SUGGEST</h1>
Hledej <input id = "vyraz" type = "text"
name = "vyraz" onkeyup = "Napoveda(event)">
<div id = "napovedaDiv"><div></div></div>
</body>
</html>
```

Podle dostupných informací bude název této funkce brzo změněn na `window.google.ac.Suggest_apply`.

Jiná bude i struktura vrácených polí.

```
window.google.ac.Suggest_apply(frameElement, "ajax",
new Array(2, "ajax", "66,400,000 results", "ajax tutorial",
"765,000 results", "ajax
    toolkit", "306,000 results", "ajax control toolkit",
    "476,000 results",
    "ajax tutorials", "1,530,000 results", "ajax examples", "258,000
    results",
    "ajax fc", "389,000 results", "ajax amsterdam",
    "1,440,000 results", "ajax
    asp.net", "1,780,000 results", "ajax framework",
    "1,610,000 results"),
new Array(""));
```